Multi-view learning-based data proliferator for boosting classification using highly imbalanced classes

Graa, Olfa; Rekik, Islem

Published in:
Journal of Neuroscience Methods

DOI:
10.1016/j.jneumeth.2019.108344

Publication date:
2019

Multi-View Learning-Based Data Proliferator for Boosting Classification Using Highly Imbalanced Classes

Olfa Graa[a,b], Islem Rekik[a,c,*]

[a] *BASIRA lab, Faculty of Computer and Informatics, Istanbul Technical University, Istanbul, Turkey*
[b] *University of Sousse, ENISo, Sousse, Tunisia*
[c] *School of Science and Engineering, Computing, University of Dundee, UK*

Abstract

-Background. Multi-view data representation learning explores the relationship between the views and provides rich complementary information that can improve computer-aided diagnosis. Specifically, existing machine learning methods devised to automate neurological disorder diagnosis using brain data provided new insights into how a particular disorder such as autism spectrum disorder (ASD) alters the brain construct. However, the performance of machine learning methods highly depends on the size of the training samples from both classes. In a real-world clinical setting, such medical data is very expensive and challenging to collect, might (i) suffer from several limitations such as imbalanced classes and (ii) have non-heterogeneous distribution when derived from multi-view brain representations.

-New Method. To the best of our knowledge, the problem of imbalanced and multi-view data classification remains unexplored in the field of *network neuroscience*. To fill this gap, we propose a *Multi-View LEArning-based data Proliferator (MV-LEAP)* that enables the classification of imbalanced multi-view representations. MV-LEAP comprises two key steps. *First*, a manifold learning-based proliferator, which enables to generate synthetic data for each view, is developed to handle imbalanced data. *Second*, a multi-view manifold data alignment leveraging tensor canonical correlation analysis is proposed to map all original and proliferated (i.e., synthesized) views into a shared subspace where their distributions are aligned for the target classification task.

*Corresponding author; Dr Islem Rekik, http://basira-lab.com/

-Results. We evaluated our method on imbalanced multi-view ASD vs normal control connectomic datasets with imbalanced classes.

-Conclusion. Overall, MV-LEAP achieved the best classification results in comparison with baseline data synthesis methods.

Keywords: imbalanced classification, multi-view data, manifold learning, data proliferator, brain network synthesis, connectomic data distribution alignment, tensor canonical correlation analysis.

---

1. Introduction

Over the past few years, remarkable advances in imaging technology have enabled researchers to apply machine learning tools to multi-view data (Zhao et al., 2017; Li et al., 2018; Sun, 2013; Zhang et al., 2019). Multi-view data provides a complementary representation of a set of samples using different types of features generated from different sources. Indeed, capturing the relationship between data views can help spot shared and discriminative traits across views, which can be utilized to boost the performance of classification frameworks. Several studies have been conducted for multi-view representation learning in different fields including computer vision for video tracking (Li et al., 2016; Ahmad and Lee, 2006), pattern recognition for image classification (Wu et al., 2016; Luo et al., 2015a; Yang et al., 2019), and text classification (Guo and Xiao, 2012; Amini et al., 2009). In particular, multi-view data have been used in diverse medical data applications such as diagnosis of diseases (Cao et al., 2019; Wang et al., 2018; Liu et al., 2017) and more specifically for autism spectrum disorder (ASD) diagnosis (Soussia and Rekik, 2017, 2018; Dhifallah et al., 2019). To non-invasively investigate how ASD affects the brain, neuroscience researchers have heavily relied on magnetic resonance imaging (MRI) to study brain connectomes also known as brain networks (Baron-Cohen et al., 1999; Sporns, 2013). The brain connectome, encoded in a network, models the relationship between a pair of anatomical regions of interest (ROIs), which is generally measured using resting-state functional MRI (rsfMRI) to derive functional brain networks or diffusion-weighted MRI (dMRI) to build structural networks (Sporns, 2013). More recently, morphological brain networks derived from T1-weighted MRI were introduced to model the relationship in morphology between brain regions in both healthy and disordered populations (Mahjoub et al., 2018; Lisowska et al., 2017, 2018; Soussia and Rekik, 2018; Dhifallah et al., 2019; Raeper

et al., 2018; Nebli and Rekik, 2019).

The brain can be represented by a single network, called single view representation, or by multiple networks, called multi-view representation. Each view captures a specific morphological trait of the brain such as the mean cortical thickness or the mean sulcal depth. Examining multi-view brain data can help us better understand both the healthy and disordered brain and eventually spot atypical brain alternations in neurological disorders. Indeed, combining different views represented by different sets of features extracted from multiple data sources provides complementary information which can boost diagnosis based solely on a single brain view representation (Li et al., 2018; Sun, 2013). However, these views are usually extracted from different modalities or using different methodologies; hence, they might lie on different domains, with heterogeneous distributions. One conventional way of jointly integrating multi-view data is to concatenate all features in a single vector. However, this is prone to over-fitting and to class overlap, and leads to the loss of complementary and correlated information which could produce erroneous results (Zhao et al., 2017; Serra et al., 2019; Luo et al., 2018; Zhao et al., 2018). Moreover, in computer-aided diagnosis (CAD) systems, the classes are usually imbalanced (Provost, 2000; Sun et al., 2009). Besides, the data distributions are usually skewed and samples drawn from different groups might overlap, making the classification of imbalanced multi-view representation challenging and prone to bias. Without loss of generality, considering we have high-dimensional multi-view connectome data where each single view has two imbalanced classes, the problem can be solved in two steps: (i) handling data imbalance by proliferating the minority class for each single view to balance both classes, and (ii) integrating the different data views by mapping and aligning them onto a shared subspace.

Learning from imbalanced data has been a challenge in machine learning over decades (Provost, 2000; Maloof, 2003; He et al., 2008; Sun et al., 2009; Li et al., 2017; Capponi and Koço, 2019; Wang et al., 2019). The most popular methods are sampling techniques such as undersampling and oversampling. Undersampling techniques are based on decreasing the number of majority class samples (Liu et al., 2009). However, removing samples might cause the loss of important information and modify the data distribution (Chawla et al., 2004). On the other hand, oversampling techniques aim to proliferate the minority class by generating synthetic samples. A landmark oversampling method is the Synthetic Minority Oversampling Technique (SMOTE) (Chawla et al., 2002). SMOTE has been widely used due to its simplicity and

robustness (Fernández et al., 2018a). This classic method is based on interpolating new synthetic samples at random distances between the k nearest neighbors.

In the last years, a few variants of SMOTE were proposed in order to improve the prediction or classification results. These variants include SMOTEBoost (Chawla et al., 2003), Borderline-SMOTE (Han et al., 2005), ADASYN (He et al., 2008), SMOTE-FRST (Ramentol et al., 2012), SDD-SMOTE (Shao et al., 2014) and Cure-SMOTE (Ma and Fan, 2017). Although, these methods have improved the classification results of imbalanced data, they present a few limitations when handling high-dimensional multi-view data such as clinical data. First, SMOTE and its derivative methods fail to select accurate nearest neighbors, due to the use of predefined distances (e.g., Euclidean distance) to search for the k nearest neighbors, which produces noisy synthetic samples. Indeed, high dimensional data are more conform to the manifold property. In the other hand, non-aligned distributions of multiple data views are not comparable, and a simple concatenation of these distributions could produce a biased classification. To address all previous limitations, we propose Multi-View LEArning based Proliferator (MV-LEAP) framework to handle multi-view data representation with data imbalance. The main contributions of MV-LEAP are twofold:

- First, in order to handle data imbalance, we propose View-Specific LEArning-based Proliferator (VS-LEAP), which proliferates training single view samples based on *learning* the similarities between data samples from the minority class. We use the sample-to-sample similarity matrix generated by multi-kernel manifold learning (ML) (Wang et al., 2017) to select the nearest neighbors and proliferate the real samples based on these selected neighbors.

- Second, to train the target classifier, we propose a joint-view alignment of all training samples (i.e., synthesized and real samples), where we learn a joint mapping of single viewed synthetic and real samples onto a shared subspace where multi-view data distributions are aligned. To this aim, we leverage tensor canonical correlation analysis (TCCA) Luo et al. (2015b) to map view-specific distributions onto a shared subspace where their correlation is maximized.

2. Proposed method

In this section, we detail each step of the proposed MV-LEAP pipeline. Figure 1 illustrates the two main steps of MV-LEAP: (a) View-Specific LEArning-based Proliferator (VS-LEAP), and (b) joint original and proliferated view alignment. In the following, we denote tensors by boldface Euler script letters (e.g., $\mathcal{X}$), matrices by boldface capital letters (e.g., $\mathbf{X}$), and scalars by lowercase letters (e.g., $\mathbf{x}$). Table 1 presents the key mathematical notations utilized in this paper.

Table 1: *Key mathematical notations used in this paper.*

Mathematical notation	Definition
n_r	number of anatomical regions of interest (ROIs)
n_v	number of views
n_f	number of connectomic features
n_s	number of original samples
n_c	number of clusters for manifold learning
n_{min}	number of minority class samples of the training data
n_{syn}	number of proliferated samples
$\mathbf{x_i}$	i-th real sample vector $\in \mathbb{R}^{n_f}$
$\tilde{\mathbf{x}}_\mathbf{i}$	i-th synthetic sample vector $\in \mathbb{R}^{n_f}$
$\mathbf{X}^\mathbf{s}_{(\mathbf{v_i})}$	i-th data matrix view for subject $s \in \mathbb{R}^{n_s \times n_f}$
$\mathbf{B}^\mathbf{s}_{(\mathbf{v_i})}$	i-th brain network view for subject $s \in \mathbb{R}^{n_r \times n_r}$
$\mathcal{T}_s = \{\mathbf{B}^\mathbf{s}_{(\mathbf{v_1})}, \mathbf{B}^\mathbf{s}_{(\mathbf{v_2})}, \ldots, \mathbf{B}^\mathbf{s}_{(\mathbf{v_q})}\}$	brain tensors of subject s with q frontal views $\in \mathbb{R}^{n_r \times n_r \times n_v}$
$\mathbf{y}$	label vector $\in \mathbb{R}^{n_s}$
$\mathbf{L}$	latent matrix $\in \mathbb{R}^{n_{min} \times n_c}$
$\mathbf{C_{xy}}$	covariance matrix of x and y
$\mathbf{E}[.]$	variance matrix
C_{min}	minority class
$\mathbf{S}$	similarity matrix $\in \mathbb{R}^{n_{min} \times n_{min}}$
$\mathbf{K}(\mathbf{x_i}, \mathbf{x_j})$	Gaussian kernel $\in \mathbb{R}^{n_f \times n_f}$ of the i-th and the j-th samples
$tr(M)$	trace of a matrix M

Given a training dataset, each sample s is represented by a tensor $\mathcal{T}_s$ of size $n_r \times n_r \times n_v$, where n_r denotes the number of cortical regions of interest (ROIs) in each hemisphere of the brain, and n_v is the number of views. Each brain network view i for subject s is encoded in a symmetric matrix $\mathbf{B}^\mathbf{s}_{(\mathbf{v_i})}$ of dimension $n_r \times n_r$ such as $\mathcal{T}_s = \{\mathbf{B}^\mathbf{s}_{(\mathbf{v_1})}, \mathbf{B}^\mathbf{s}_{(\mathbf{v_2})}, \ldots, \mathbf{B}^\mathbf{s}_{(\mathbf{v_q})}\}$, where q is the number of frontal views of the tensor. Each element $\mathbf{b_{ij}}$ of the matrix $\mathbf{B}^\mathbf{s}_{(\mathbf{v_i})}$ encodes the relationship between pairs of ROIs. For each subject s, we vectorize the upper off-diagonal triangular part of each matrix $\mathbf{B}^\mathbf{s}_{(\mathbf{v_i})}$ in $\mathcal{T}_s$, to obtain a connectivity feature vector of size $n_f = (n_r \times (n_r - 1)/2)$. Then, the extracted view-specific feature vectors are vertically concatenated to form a data matrix $\mathbf{X}^\mathbf{s}_{(\mathbf{v_i})}$ of size $n_s \times n_f$, where n_s is the total number of

subjects, and n_f is the size of each row feature vector. $\mathbf{X}^{\mathbf{s}}_{(\mathbf{v_i})}$ is a matrix of high dimension where $n_f \gg n_s$.

We first propose to balance the data using the View-Specific LEArning-based Proliferator (VS-LEAP). Second, in order to preserve the complementary and correlated information among both real and proliferated views, we propose to learn a low dimensional representation based TCCA, where the data distributions are mapped and aligned into a shared low-dimensional subspace.

2.1. View-Specific LEArning-based Proliferator (VS-LEAP)

Fig. 1–A illustrates the proposed proliferator based multi-kernel manifold learning. The proliferator aims to generate new synthetic samples of the minority class to balance the training data. Particularly, we propose to generate synthetic data using the SMOTE algorithm. The main idea of SMOTE is to interpolate points between nearest neighbors, by operating in the features space. **Fig.** 2 illustrates the proliferation of real samples using SMOTE. For each sample $\mathbf{x_i}$ of the minority class C_{min}, SMOTE generates a synthetic sample $\tilde{\mathbf{x}}$ at a random distance on the line segment connecting $\mathbf{x_i}$ to its nearest neighbor $\mathbf{x_j} \in \{C_{min} \cup kNN_i\}, j \in \{1, \ldots, k\}$. $\mathbf{x_j}$ is randomly chosen from the k Nearest Neighbors (kNN) set, depending upon the amount of over-sampling required. Each synthetic sample $\tilde{\mathbf{x}}$ is generated as follows:

$$\tilde{\mathbf{x}} = \mathbf{x_i} + (\mathbf{x_j} - \mathbf{x_i}) \times \delta \tag{1}$$

where δ is a random number between 0 and 1.

Over the years, SMOTE has inspired many researchers and some variants were proposed to improve the prediction or classification results (Fernández et al., 2018b). Nonetheless, SMOTE and its variant use predefined distances, such as the Euclidean distance, the Manhattan distance and the Minkowski distance, to search for the kNNs. These metrics can cause the proliferation of noisy real samples, due to the incompatibility of predefined distance with the high dimensional data distribution where data may have a non-linear relationship. For this reason, we propose to use a manifold technique that better fits the statistical distribution of the data to proliferate real samples from the minority class of training samples for each view individually in two steps (**Fig.** 1–A). First, for each training sample, we project the data samples of the minority class on the manifold space to detect the kNNs set. Second, we use the KNNs set to generate synthetic samples of the minority class in

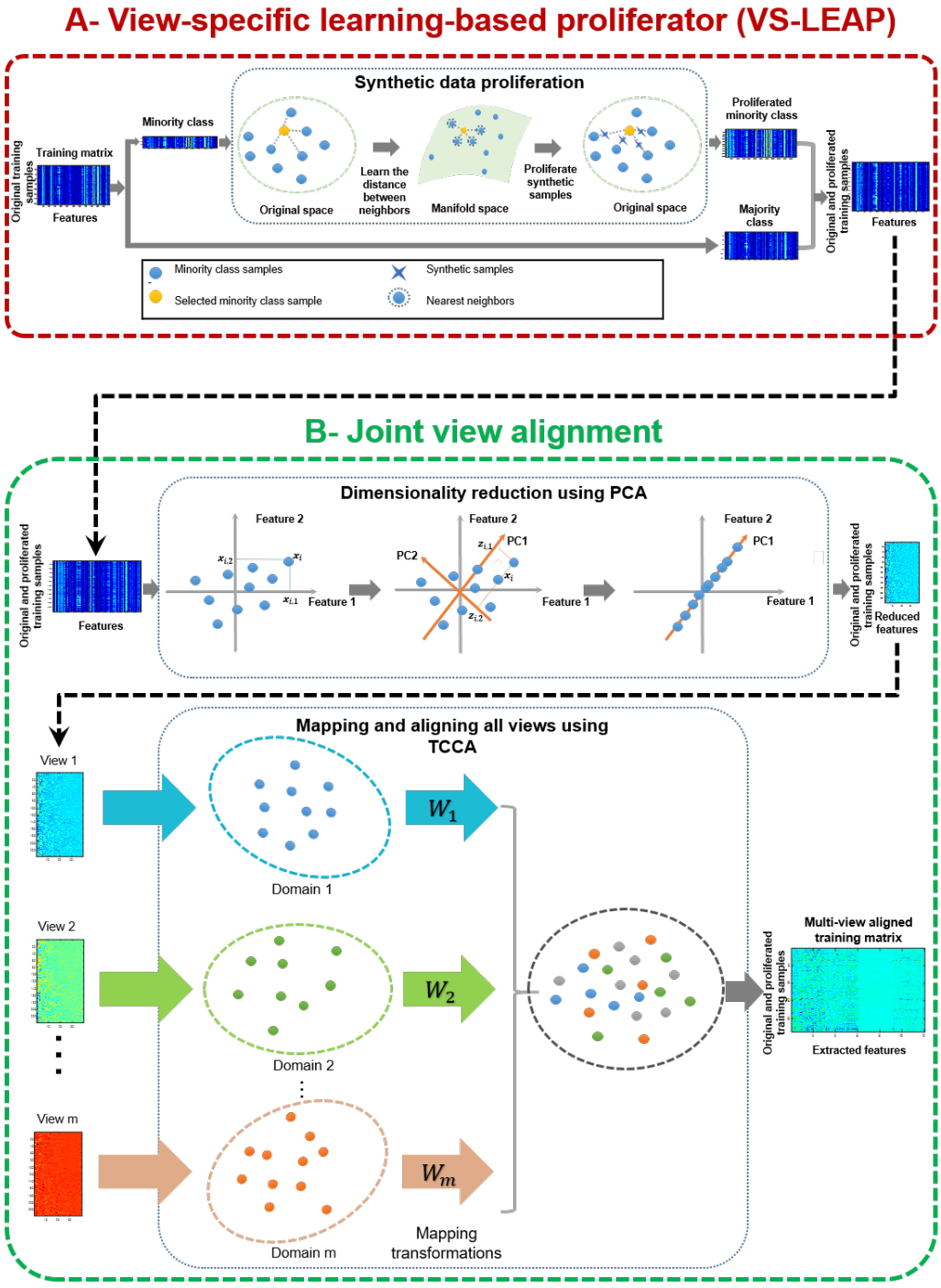


Figure 1: *Illustration of the proposed MV-LEAP framework.* (A) describes the View-Specific LEArning-based proliferator (VS-LEAP) applied to one data view. First, we proliferate the minority class by generating new synthetic samples using manifold learning. (B) presents the joint view alignment of proliferated and original data using tensor canonical correlation analysis (TCCA). First, we use principle component analysis (PCA) to reduce the dimensionality of the training data in order to prepare the training data for the TCCA. Second, we map all views onto a shared subspace by projecting them using their learned TCCA transformation matrices (or mappings) $\{\mathbf{W_1}, \mathbf{W_2}, \ldots, \mathbf{W_m}\}$. This step aligns the distributions of all transformed views in a low-dimensional shared subspace. This produces a final training matrix concatenating the extracted multi-view aligned features of both original and proliferated samples in the learned low-dimensional shared subspace for training the target classifier.

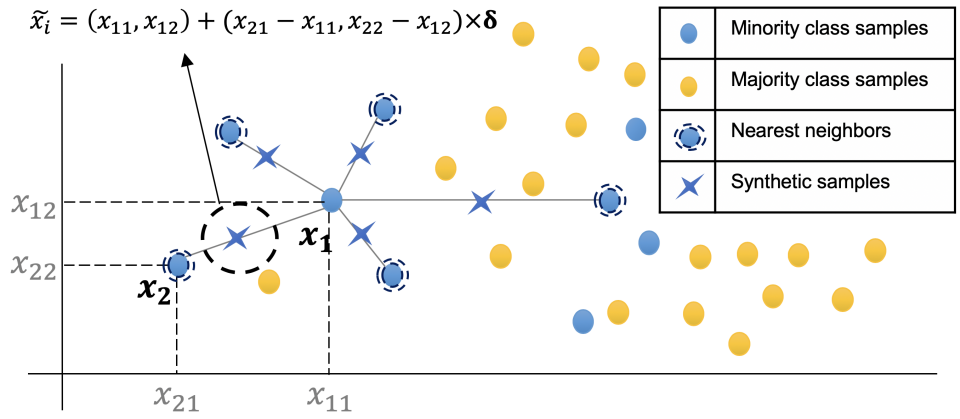


Figure 2: *Illustration of sample proliferation in 2D space using the traditional SMOTE (Chawla et al., 2002).* $\mathbf{x_1}$ is a real sample with coordinates (x_{11}, x_{12}). The synthetic samples are generated at a random distance between $\mathbf{x_1}$ and its k nearest neighbors. Given $\mathbf{x_2}$ one of the 5 nearest neighbors of $\mathbf{x_1}$, with coordinates (x_{21}, x_{22}), the synthetic sample is calculated using the Euclidean distance as follows: $\tilde{x}_i = (x_{11}, x_{12}) + (x_{21} - x_{11}, x_{22} - x_{12}) \times \delta$, where δ is a random number between 0 and 1.

the original space. For the first step, we use a multi-kernel manifold learning based on the Single-cell Interpretation via Multi-kernel LeaRning (SIMLR) algorithm (Wang et al., 2017). The main idea of SIMLR consists of learning sample-to-sample similarity measure. Given a data matrix $\mathbf{X^s_{vi}} \in \mathbb{R}^{n_s \times n_f}$ as input, SIMLR estimates proper weights for multiple Gaussian kernels which represent different measures of sample-to-sample distance. The learned kernels are combined to build a similarity matrix $\mathbf{S} \in \mathbb{R}^{n_{min} \times n_{min}}$.

A Gaussian kernel is defined as:

$$\mathbf{K}\left(\mathbf{x}_i, \mathbf{x}_j\right) = \frac{1}{\sqrt{2\pi}\sigma\left(\mu_i, \mu_j\right)/2} e^{-\frac{\left|\mathbf{x}_i - \mathbf{x}_j\right|^2}{\sigma\left(\mu_i + \mu_j\right)^2/2}} \tag{2}$$

where

- $\mathbf{x}_i$ and $\mathbf{x}_j$ are the i-th and the j-th feature vectors.
- $\mu_i = \frac{\sum_{l \in kNN(\mathbf{x}_i)} |\mathbf{x}_i - \mathbf{x}_j|}{k}$, where $kNN(\mathbf{x}_i)$ defines the top k neighboring samples of the i-th sample.

The general form of the distance between a sample $\mathbf{x}_i$ and a sample $\mathbf{x}_j$ is: $d\left(\mathbf{x}_i, \mathbf{x}_j\right) = 2 - 2\sum_l w_l K_l\left(\mathbf{x}_i, \mathbf{x}_j\right)$, where $\mathbf{K}_l(\mathbf{x}_i, \mathbf{x}_j)$ is an individual kernel

modeling the similarity between the i-th and the j-th samples, and w_l is a weight value assigned to the l^{th} kernel $\mathbf{K}_l$. The optimization framework of SIMLR (Wang et al., 2017) aims to minimize the following energy functional:

$$\begin{gathered}\min_{\mathbf{S},\mathbf{L},\mathbf{w}} -\sum_{i,j,l} \mathbf{w}_l \mathbf{K}_l\left(\mathbf{x}_i, \mathbf{x}_j\right) \mathbf{S}_{ij} + \beta\|\mathbf{S}\|_F^2 + \gamma \operatorname{tr}\left(\mathbf{L}^t\left(\mathbf{I}-\mathbf{S}\right)\mathbf{L}\right) + \rho\sum_l \mathbf{w}_l \log \mathbf{w}_l \\ \text{subject to } \mathbf{L}^T\mathbf{L} = \mathbf{I}_C \sum_l \mathbf{w}_l = 1, \mathbf{w}_l \geq 0, \sum_l \mathbf{S}_{ij} = 1, \text{ and } \mathbf{S}_{ij} \geq 0\end{gathered} \tag{3}$$

The first term $\sum_{i,j,l} \mathbf{w}_l \mathbf{K}_l\left(\mathbf{x}_i, \mathbf{x}_j\right) \mathbf{S}_{ij}$ invokes the relationship between a kernel of the i-th and the j-th samples multiplied by its weight $\mathbf{w}_l\mathbf{K}_l(\mathbf{x}_i, \mathbf{x}_j)$, and the similarity $\mathbf{S}_{ij}$. It requires that if the distance between two samples is large, then, the learned similarity $\mathbf{S}$ between them should be small.

The second term $\beta\|\mathbf{S}\|_F^2$ is a regularization term that avoids the similarity matrix S from becoming an identity matrix. β is a non-negative tuning parameter and $\|\mathbf{S}\|_F$ is the Frobenius norm of $\mathbf{S}$.

The third term $\gamma \operatorname{tr}\left(\mathbf{L}^t\left(\mathbf{I}_N - \mathbf{S}\right)\mathbf{L}\right)$, enforces the low-rank structure of $\mathbf{S}$. γ is a non-negative tuning parameter, $tr(.)$ is the trace of the matrix, $(\mathbf{I}_N - \mathbf{S})$ is the Laplacian graph where $\mathbf{I}$ is an identity matrix of size $n_{min} \times n_{min}$, and $\mathbf{L}$ is an auxiliary low-dimensional matrix of size $n_{min} \times n_c$, where n_c is the number of clusters.

The fourth term $\rho\sum_l \mathbf{w}_l \log \mathbf{w}_l$ is a regularization term that avoids the selection of a single kernel.

The optimization framework of SIMLR aims to solve for three variables: (1) the similarity matrix $\mathbf{S}$, (2) the weight vector $\mathbf{w}$, and (3) the latent matrix $\mathbf{L}$. Next, we use the similarity matrix $\mathbf{S}$ to capture the kNNs of the minority class of the training samples. When performing SIMLR, we obtain a similarity matrix $\mathbf{S}$ of dimension $n_{min} \times n_{min}$, where n_{min} is the size of the minority class. Once $\mathbf{S}$ is learned, we select the k-nearest neighbors of each minority class sample. The kNNs for a sample i are the k most similar samples, such as the similarity between the samples i and j is stronger as $\mathbf{S}_{(i,j)}$ approaches 1. Next, for each training minority class sample, we retain only $p_{OS}\%$ of the top k-nearest neighbors, such as the retained synthetic sample are randomly selected. We interpolate the synthetic samples between the real samples and their selected kNN. The proposed manifold learning-based

proliferation process is summarized in Algorithm 1.

Algorithm 1: Synthetic data proliferation

Data: data matrix of the minority class of the training data $\mathbf{X_{min}}$, percentage of oversampling p_{OS} such that $p_{OS} \geqslant 100$, number of nearest neighbors for SMOTE K_{SMOTE} such that $K_{SMOTE} \geqslant 2$, number of nearest neighbor for manifold learning (ML) K_{ML} such that $K_{ML} \geqslant 2$.

Result: proliferated minority class data matrix $\mathbf{X_{prolif}}$

begin

/* Initialization */

Ind = matrix of all the index of the nearest neighbors of $\mathbf{X}_{min}$;

S = similarity matrix;

/* learn the K_{ML} nearest neighbors */

$\mathbf{S} \leftarrow$ `manifoldLearing`$(\mathbf{X}_{min}, K_{ML})$;

for $i = 1$ **to** n_{min} **do**

  for $j = 1$ **to** K_{SMOTE} **do**

    Ind$[\mathbf{i}, \mathbf{j}]$ = the index of the j-th nearest neighbor of the i-th sample from **S**;

/* Proliferate samples */

for $l = 1$ **to** n_{min} **do**

  $\mathbf{ind}_{retain}$ = retain only $p_{OS}\%$ out of k nearest samples from **Ind**;

  δ = random number between 0 and 1;

  $\bar{\mathbf{x}}(l) = \mathbf{x}(l) + (\mathbf{x}(\mathbf{ind}_{retain})\mathbf{x}(l)) \times \delta$;

$\mathbf{X_{prolif}} = \{\mathbf{X}_{min}, \bar{\mathbf{X}}_{\mathbf{min}}\}$

Next, using both original and proliferated samples for each view, we will jointly align all data views to a shared low-dimensional subspace where a joint set of embeddings is learned (**Figure** 1–B). Ultimately, the proliferated aligned data will be used to train a classification model.

2.2. Joint view alignment

Preliminary view-specific dimension reduction. Our input data is of high dimensionality. The size of the feature vector n_f is greater than the size of the proliferated samples n_{syn} ($n_f \gg n_{syn}$). This can be memory-consuming when aiming to learn a joint embedding of all views into a shared subspace using joint dimension reduction techniques such as TCCA (Luo et al., 2015b). For this reason, the high dimension of the data needs to be

reduced. There are two types of dimension reduction techniques: feature selection and feature extraction. Feature selection techniques, such as (Roffo et al., 2015), aim to select the most relevant data and remove the less important data. This can lead to a loss of information. On the other hand, feature extraction techniques aim to project the data into a new subspace while preserving the maximum data information. To solve our problem, we propose to use one of the most relevant feature extraction methods named principal component analysis (PCA) (Abdi and Williams, 2010). PCA extracts important information from data by maximizing the variance of the projected data onto a lower dimensional subspace. It transforms the original features into a set of new orthogonal and linearly uncorrelated features known as principal components (PC). The principal components are computed using the eigenvectors and the eigenvalues of the covariance matrix of the data. When we apply PCA on each view independently, we get n_v new representations of the data encoded in low-dimensional aligned matrices with balanced classes. Next, we horizontally concatenate the view-specific low-dimensional matrices to form one final multi-view aligned training matrix using the proposed view alignment technique (**Figure** 1–B).

Joint view alignment. We assume that data views are nested in different domains. For this reason, we propose to align each view onto a shared low-dimensional subspace. Joint view alignment is illustrated in **Figure.** 1–B. To solve this problem, we use the tensor canonical correlation analysis (TCCA) Luo et al. (2015b). TCCA is a generalization of canonical correlation analysis (CCA) to handle multi-view data representation. Specifically, CCA is a method that measures the linear relationship between two high-dimensional datasets or data views. CCA aims to find the optimal projection for each set where the correlation between the two sets is maximal.

Given two vectors $\mathbf{x}_1 \in \mathbb{R}^{m_1}$ and $\mathbf{x}_2 \in \mathbb{R}^{m_2}$, their linear projections onto $\mathbf{w}_1$ and $\mathbf{w}_2$, respectively, are $\mathbf{z}_1 = \mathbf{w}_1^T \mathbf{x}_1$ and $\mathbf{z}_2 = \mathbf{w}_2^T \mathbf{x}_2$. CCA aims to find the optimal pair of canonical vectors $\mathbf{w}_1$ and $\mathbf{w}_2$, where the correlations of the projected vectors is maximized. The optimization problem of CCA is formalized as follows:

$$\text{argmax}_{\{\mathbf{w}_1,\mathbf{w}_2\}}\, \rho = \text{corr}\left(\mathbf{z}_1, \mathbf{z}_2\right) = \frac{E\left[\mathbf{z}_1 \mathbf{z}_2^T\right]}{\sqrt{E\left[\mathbf{z}_1 \mathbf{z}_1^T\right] E\left[\mathbf{z}_2 \mathbf{z}_2^T\right]}} = \frac{\mathbf{w}_1^T \mathbf{C}_{\mathbf{x}_1\mathbf{x}_2} \mathbf{w}_2}{\sqrt{\mathbf{w}_1^T \mathbf{C}_{\mathbf{x}_1\mathbf{x}_1} \mathbf{w}_1 \mathbf{w}_2^T \mathbf{C}_{\mathbf{x}_2\mathbf{x}_2} \mathbf{w}_2}} \tag{4}$$

Where ρ is the canonical correlation. Each pair of transformed vectors (e.g., $\mathbf{w}_1$ and $\mathbf{w}_2$) is orthogonal to the previous ones and is associated with

$\{\rho_1, \rho_2, \ldots, \rho_q\}$ where $q < \min(m_1, m_2)$. From a probabilistic viewpoint, two random samples $\mathbf{x}$ and $\mathbf{y}$ can be represented by a common sample $\mathbf{z} \in \mathbb{R}^q$ such as the affine transformations are defined as probabilistic functions or likelihoods $P(\mathbf{x}|\mathbf{z})$ and $P(\mathbf{y}|\mathbf{z})$. By maximizing the likelihood estimation, CCA estimates the optimal mapping from source domains to the shared target domain.

The dimensionality of the new projected sample is equal to or less than the smallest dimension of the two samples. For this reason, CCA is not only a pairwise data concatenation method, but also a dimension reduction strategy. Compared to ordinary correlation analysis, CCA is a robust and flexible method since it is invariant to affine transformation of samples. However, while CCA explores only the pairwise correlation (low-order correlation) between two datasets (or data views), TCCA enables to explore all correlations (low- and high-order correlations) between all views *simultaneously*. It aims to directly maximize the canonical correlation of multi-view representation by analyzing the covariance tensor. The optimization problem of TCCA is formulated as follows:

$$\begin{gathered} \operatorname{argmax}_{\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_q\}} \rho = \operatorname{corr}(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_q) \\ \text{Subject to } \mathbf{z}_p^T \mathbf{z}_p = 1, p = 1, \ldots, q \end{gathered} \tag{5}$$

where $\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_q\}$ denotes the canonical vectors.

It has been proven that this maximization problem is equivalent to finding the best rank-1 approximation of the data covariance tensor Luo et al. (2015b). This problem can be solved using alternating least squares (ALS) algorithm (Kroonenberg and De Leeuw, 1980; Comon et al., 2009). Given a set of m data views $\mathcal{T}_m = \left\{\mathbf{X}^s_{(v_1)}, \mathbf{X}^s_{(v_2)}, \ldots, \mathbf{X}^s_{(v_m)}\right\}$, where each view is represented by a data matrix of n_s subjects and n_f features such as $\mathbf{X}^s_{(v_p)} = \left[\mathbf{x}_{p1}, \mathbf{x}_{p2}, \ldots, \mathbf{x}_{pn_f}\right] \in \mathbb{R}^{n_f \times n_s}, p = 1, \ldots, n_s$. TCCA proceed as follows:

1. Calculating a covariance tensor $\mathcal{C}_m$ in order to explore the correlation information between views. The covariance tensor is calculated as: $\mathcal{C}_m = \frac{1}{n_s} \sum_{i=1}^{n_s} \mathbf{x}_{1i} \circ \mathbf{x}_{2i} \circ \ldots \circ \mathbf{x}_{mi}$, where $\circ$ denotes the tensor outer product.
2. Computing the transformation matrices $\{\mathbf{W}_1, \mathbf{W}_2, \ldots, \mathbf{W}_m\}$ corresponding to the m views, respectively, by approximating the covariance tensors with a set of rank-1 tensors.

3. Mapping each original matrix $\left\{\mathbf{X}^s_{v_1}, \mathbf{X}^s_{v_2}, \ldots, \mathbf{X}^s_{v_m}\right\}$ to a low dimensional matrix $\{\mathbf{Z}_1, \mathbf{Z}_2, \ldots, \mathbf{Z}_m\}$, using the transformation matrices $\{\mathbf{W}_1, \mathbf{W}_2, \ldots, \mathbf{W}_m\}$. This allows the projection of the data onto the common subspace.
4. Concatenation of all the low-dimensional matrices to obtain the final matrix $\mathbf{Z}$.

3. Results

Evaluation dataset. To evaluate our proposed framework, we used 150 subjects (50 ASD and 150 NC) from the Autism Brain Imaging Data Exchange (ABIDE I) public dataset [1], each with structural T1-w MR image. Table 2 displays the data distribution. Since ASD is prevalent in male subjects (Christensen et al., 2018), our cohort only included males. For each subject, both right and left cortical hemispheres (RH and LH) were reconstructed using FreeSurfer (Fischl, 2012). Then, each cortical hemisphere was split into 35 cortical regions using Desikan-Killiany atlas (Desikan et al., 2006). Four cortical attributes were assigned to each vertex on the cortical surface using FreeSurfer. These attributes are the maximum principal curvature, the cortical thickness, the sulcal depth, and the average curvature. Based on these attributes, four morphological networks (also called views) were generated for each subject. For each cortical view, the strength of the connectivity between two ROIs i and j is computed as the absolute difference between the average cortical attribute in ROI i and the average cortical attribute in ROI j (Mahjoub et al., 2018; Lisowska et al., 2018; Dhifallah et al., 2019).

Table 2: *Table of data distribution*

Dataset	ASD	NC
Number of subjects	50	150
Mean age	18.14	17.91

To investigate the properties of our data distributions, we plotted the histogram of the majority and the minority classes for each view of the right and the left hemisphere, as illustrated in **fig.** 3. All graphs represent positively skewed distributions. Besides, the data distribution of the minority class and the majority class overlap, which challenges the learning of high-performing

[1] http://preprocessed-connectomes-project.org/abide/

classification models. For connectomic view 2 derived from cortical thickness, we notice that the connectomic feature distribution is bimodal for both left and right hemispheres.

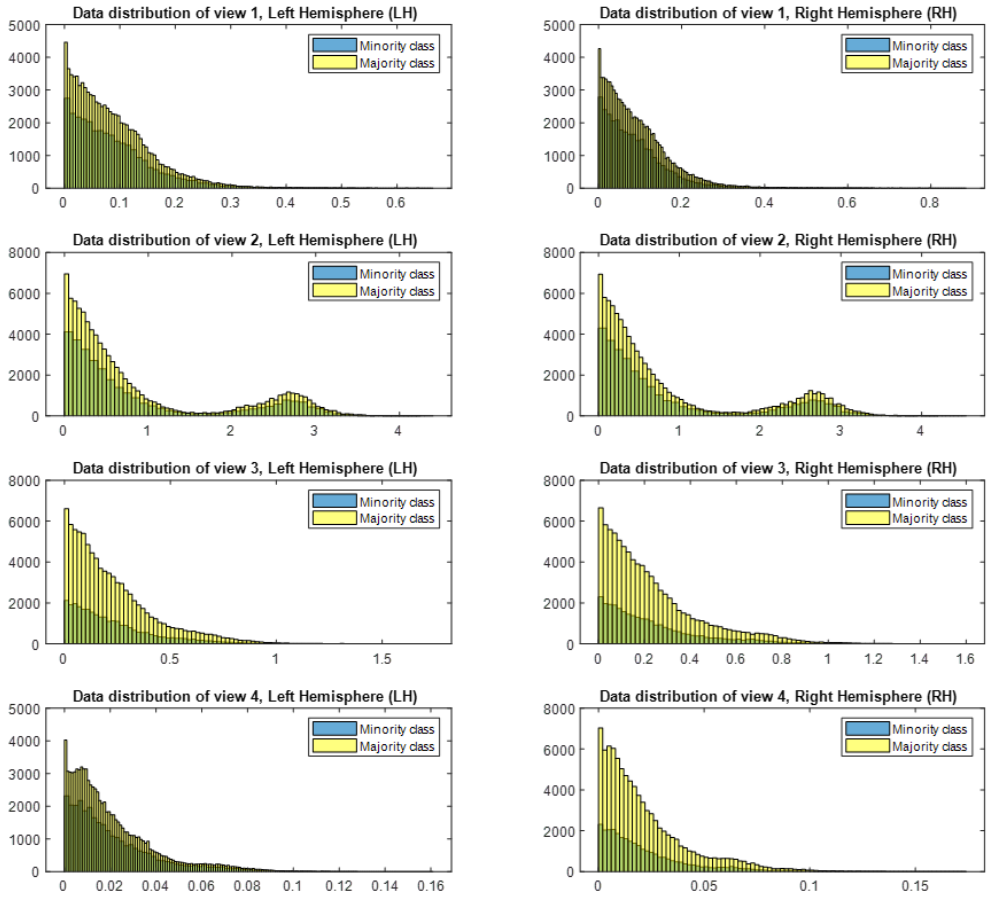


Figure 3: *Data distribution of each morphological connectomic view for the left and right hemisphere of the brain.* View 1: maximum principal curvature. View 2: cortical thickness. View 3: sulcal depth. View 4: average curvature.

Methods parameters. For SMOTE parameters, we empirically set the number of nearest neighbors to $k_{SMOTE} = 5$. For SIMLR parameters, using nested cross-validation (CV), we set the number of clusters $c = 2$, and the number of nearest neighbors to $k_{ML} = 5$. For TCCA parameters, we

set the regularization trade-off factor $\epsilon = 0.5$, and the rank of the tensor for the canonical polyadic decomposition using the alternating least squares algorithm (CP-ALS) $R = 30$.

Performance measures. To evaluate the performance of our method, we used stratified 5-fold cross-validation (5-fold-CV). Stratified 5-fold-CV is used for imbalanced data in order to have the same data distribution of classes across all folds (**Figure** 4).

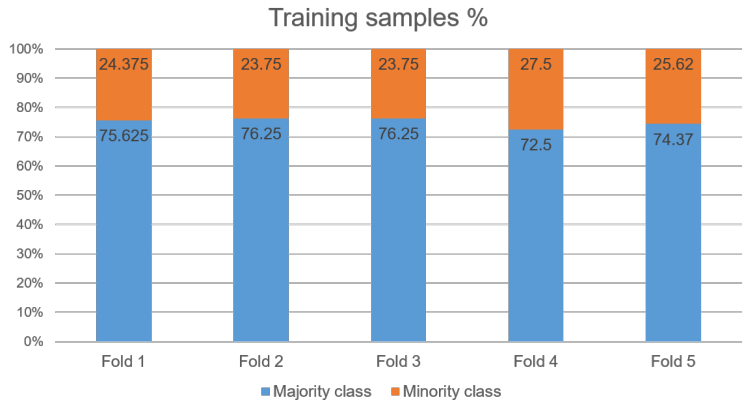


Figure 4: *Data distribution of the minority and the majority classes in each fold.* The orange color shows the percentage of minority class samples in each fold whereas the blue color displays the percentage of the majority class samples. The data distribution is largely stable across folds (standard deviation equal to 1.42).

The data samples are randomly divided into 5 folds: four are used for training and one is left out for testing. The training samples are, first, used as input to the MV-LEAP pipeline then to train a linear support vector machine (SVM) classifier. The testing samples are used to evaluate the SVM classification model. We evaluated the model using the confusion matrix, which estimates the overall accuracy and the F-measure. The overall accuracy is an evaluation metric of the overall performance of the classification model. It is defined as the number of correctly classified samples divided by the total number of samples as follows:

$$OverallAccuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{6}$$

where TP and TN represent the number of positive samples and negative samples, respectively, which were correctly classified, and FP and FN denote the number of positive samples and negative samples, respectively, incorrectly classified. However, the overall accuracy is not a precise metric for

the evaluation of imbalanced data. If the model is weak at predicting the minority class, but good at predicting the majority class, then the number of the correct prediction will be larger than the number of incorrect predictions. Thus, the overall accuracy will be high which leads to a biased evaluation of the performance of the classifier. For this reason, we use the F-measure and the balanced accuracy as a complementary metrics to the overall accuracy. F-measure is the harmonic mean of precision (PPV) and sensitivity (TPR). It is calculated as follows:

$$\begin{aligned} F- \text{ measure } &= 2\frac{TPR \cdot PPV}{TPR+PPV} = \frac{2TP}{2TP+FP+FN} \\ \text{where PPV } &= \frac{TP}{TP+FP} \text{ and } TPR = \frac{TP}{TP+FN} \end{aligned} \tag{7}$$

We also use balanced accuracy (BACC), which is commonly used for evaluating imbalanced classification models. It is defined as the average of correctly classified samples of each class individually and calculated as follows:

$$\begin{aligned} \text{BACC} &= (\frac{TP}{TP+FN} + \frac{TN}{FP+TN})/2 \\ &= \frac{(TPR+TNR)}{2} \end{aligned} \tag{8}$$

where TNR is the specificity.

In addition, we used the receiver operating characteristic (ROC) curve as evaluation metric for the binary classification. ROC curve is a probability plot. It illustrates the relationship between the sensitivity, also known as the recall or the true positive rate (TPR) and the specificity. For example, if the sensitivity increases, the specificity decreases. Roc curve plots TPR against the false positive rate (FPR), which are calculated as follows:

$$TPR = \frac{TP}{TP+FN} \tag{9}$$

$$FPR = 1 - \text{specificity} = \frac{TN}{TN+FP} \tag{10}$$

Evaluation and comparison methods. To evaluate our framework MV-LEAP, we compared its performance with baseline methods. First, we compared the proposed *learning-based* sample proliferator with the traditional SMOTE and ADASYN methods. Then, we evaluated the use of TCCA by comparing it to a simple concatenation of the views without joint view alignment.

1. **Comparison with proliferation methods.** We compared the proposed SMOTE proliferator based on manifold learning (SMOTE+ML+SVM) with the following baseline methods: **(a)** linear support vector machine (SVM), **(c)** the traditional SMOTE with SVM (SMOTE+SVM), **(c)** ADASYN with SVM (ADASYN+SVM), and **(d)** ADASYN based on manifold learning (ADASYN+ML+SVM). SVM is a discriminative classifier that learns a classification model using supervised training data. It categorizes samples using an optimal hyperplane. ADASYN is a proliferation method based on the SMOTE algorithm. Unlike SMOTE, ADASYN calculates the number of synthetic samples to be generated for each minority class entity based on a density distribution. For a fair comparison, we set the number of nearest neighbors $k = 5$ across all the methods. We used four single view representations for each cortical hemisphere.

    - **Accuracy measurement.** For the left hemisphere, SVM achieved the highest accuracy for all single view representations, as shown in **Figure** 5–LH. Notably, our method achieved the second highest accuracy for 3 views (views number 2, 3, and 4). The second highest accuracy for view 1 was achieved by ADASYN+ML+SVM. For the right hemisphere, SVM achieved the highest accuracy for all single view representations, as shown in **Figure** 5–RH. Our method achieved the second highest accuracy for one view (view number 4). ADASYN+SVM achieved the second highest accuracy for views number 2 and 3. The second highest accuracy for view 1 was achieved by SMOTE+SVM.

    - **F-measure.** For the left hemisphere, when proliferating samples from the minority class, our method achieved the highest F-measure value for 3 views (views 2, 3, and 4) (**Figure** 6). Excluding SVM which is overfitting the non-proliferated data, the best highest F-measure value for view 1 was achieved by ADASYN+SVM+ML, which leveraged the proposed idea of learning the similarities between samples to guide the data proliferation step.

      For the right hemisphere, our method outperformed benchmark proliferation-based methods for 3 views (views 1, 3, and 4). The highest F-measure value for view 2 was achieved by ADASYN+SVM.

    - **Balanced accuracy.** For the left hemisphere, our method achieved

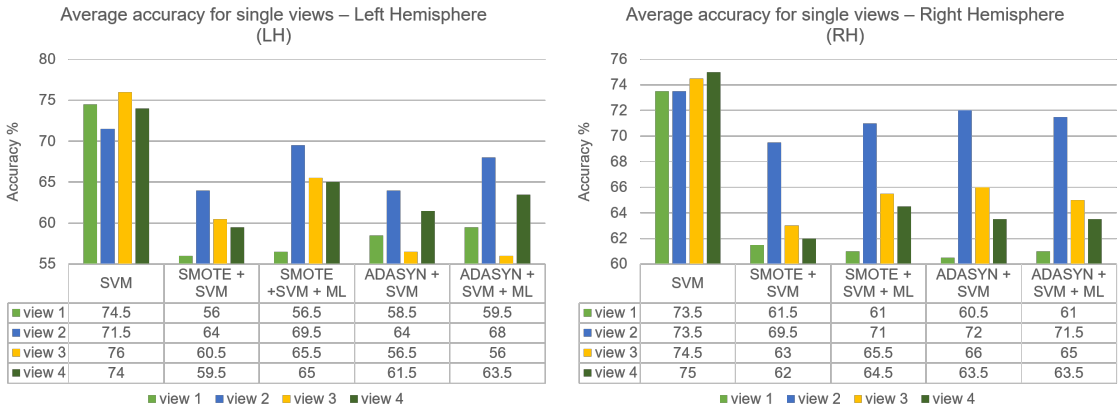


Figure 5: *Comparison of ASD/NC classification accuracy of our method (SMOTE+ML+SVM) and comparison methods (SVM, SMOTE+SVM, ADASYN+SVM, and ADASYN+ML+SVM), on each single view representation.* (Left panel) Classification accuracy for the left hemisphere. (Right panel) Classification accuracy for the right hemisphere. SVM is typically trained using majority and minority classes without any data proliferation. Its outperformance can be explained by data overfitting.

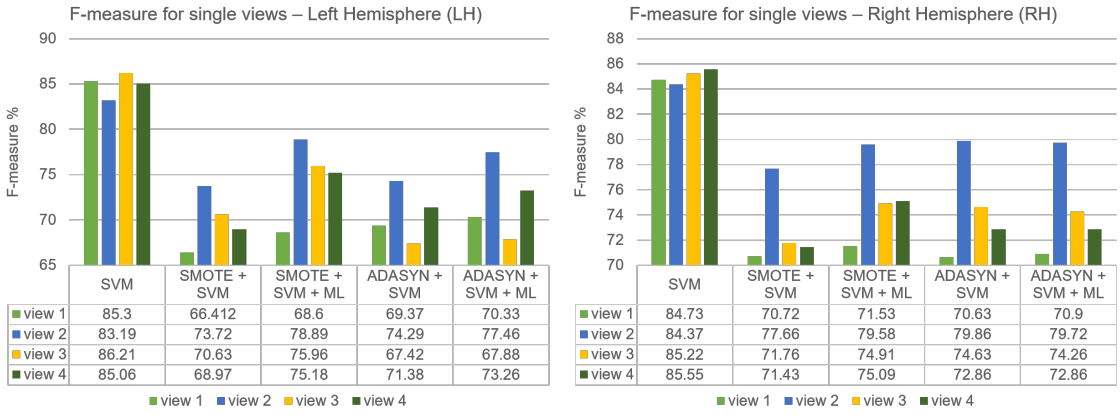
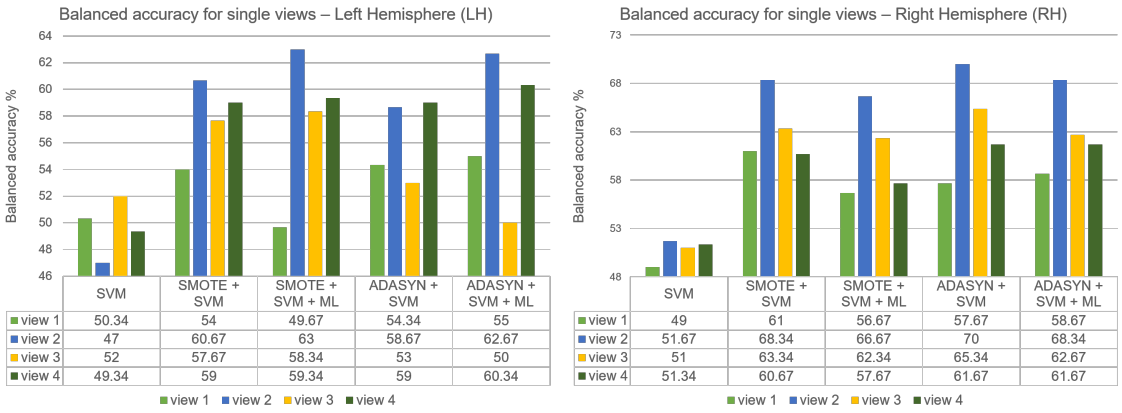


Figure 6: *Comparison of the F-measure of the ASD/NC classification using F-measure of our method (SMOTE+ML+SVM) and comparison methods (SVM, SMOTE+SVM, ADASYN+SVM, and ADASYN+ML+SVM), on each single view representation.* (Left panel) F-measure for the left hemisphere. (Right panel) F-measure for the right hemisphere. SVM is typically trained using majority and minority classes without any data proliferation. Its outperformance can be explained by data overfitting.

the highest balanced accuracy for 2 single views (views 2 and 3) as shown in **Figure** 7. ADASYN+SVM+ML achieved the highest balanced accuracy for the two other views (views 1 and 4). Notably, proliferation methods based manifold learning achieved higher results than simple proliferation methods.

For the right hemisphere, ADASYN+SVM achieved the highest balanced accuracy for 3 views (views 2, 3, and 4). SMOTE+SVM achieved the highest result for view 1. Overall, SVM achieved the lowest results for both left and right hemispheres.



	SVM	SMOTE + SVM	SMOTE + SVM + ML	ADASYN + SVM	ADASYN + SVM + ML
view 1	50.34	54	49.67	54.34	55
view 2	47	60.67	63	58.67	62.67
view 3	52	57.67	58.34	53	50
view 4	49.34	59	59.34	59	60.34

	SVM	SMOTE + SVM	SMOTE + SVM + ML	ADASYN + SVM	ADASYN + SVM + ML
view 1	49	61	56.67	57.67	58.67
view 2	51.67	68.34	66.67	70	68.34
view 3	51	63.34	62.34	65.34	62.67
view 4	51.34	60.67	57.67	61.67	61.67

Figure 7: *Comparison of ASD/NC classification results using balanced accuracy of our method (SMOTE+ML+SVM) and comparison methods (SVM, SMOTE+SVM, ADASYN+SVM, and ADASYN+ML+SVM), on each single view representation.* (Left panel) Classification accuracy for the left hemisphere. (Right panel) Classification accuracy for the right hemisphere. SVM is typically trained using majority and minority classes without any data proliferation.

2. **Comparison using TCCA.** To evaluate the benefit of using TCCA for mapping and aligning different data views, we compared the performance of the designed MV-LEAP combining VS-LEAP and TCCA, with methods that use a simple concatenation of multi-view feature vectors without alignment. The comparison methods include: **(a)** SMOTE+PCA, and **(c)** SMOTE+ML+PCA. In a first experiment, we evaluated all methods on datasets generated using different combinations of three connectomic morphological brain views. In a second experiment, we evaluated all methods on a dataset composed of the four brain views.

   - **Evaluation on three views.** We evaluated MVLEAP (i.e., SMOTE+ML+PCA+TCCA) and the baseline methods (SMOTE+PCA

and SMOTE+ML+PCA) on three brain connectomic views for both left and right hemispheres. For both left and right hemispheres, our MV-LEAP classification framework consistently outperformed all comparison methods across different datasets composed of 3 views (**Figure** 8).

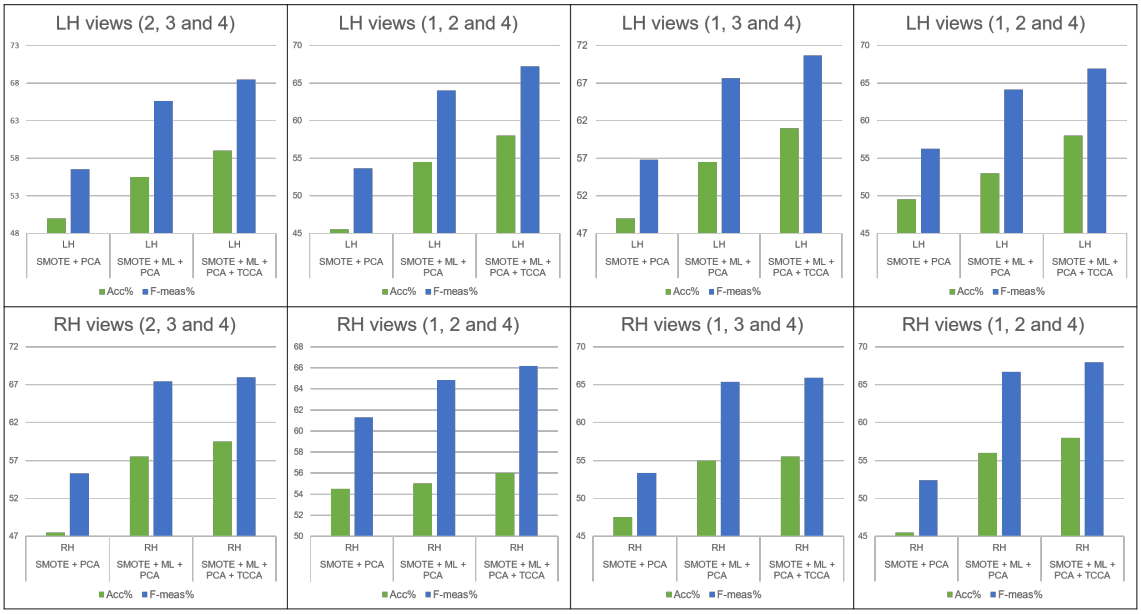


Figure 8: *Evaluation of the ASD/NC classification using the accuracy and the F-measure metrics for the left and the right hemispheres for different combinations of three morphological connectomic views.* View 1: maximum principal curvature. View 2: cortical thickness. View 3: sulcal depth. View 4: average curvature.

- **Evaluation on four views.** We compared the ASD/NC classification of our framework MV-LEAP (SMOTE+ML+PCA+TCCA) with the comparison methods (SMOTE+PCA) and (SMOTE+ML+PCA) on a dataset composed of four brain views. We evaluated the performance of our method using the overall accuracy and the F-measure, as demonstrated in **Figure** 9.

  Remarkably, for both left and right hemispheres MV-LEAP based classifier achieved the best results in distinguishing between autistic and typical subjects. In addition, we evaluated our method based on the ROC curve plots in **Figure** 10. Clearly, our method achieved the highest AUC, for both left and right hemispheres. According to these results, MV-LEAP outperformed baseline meth-

ods.

Furthermore, we calculated the performance standard deviation (STD), for each method, by repeating the classification 10 times. The error bars in **Figure** 9 represent the calculated STD. Overall, MV-LEAP achieved more stable results with the lowest STD.

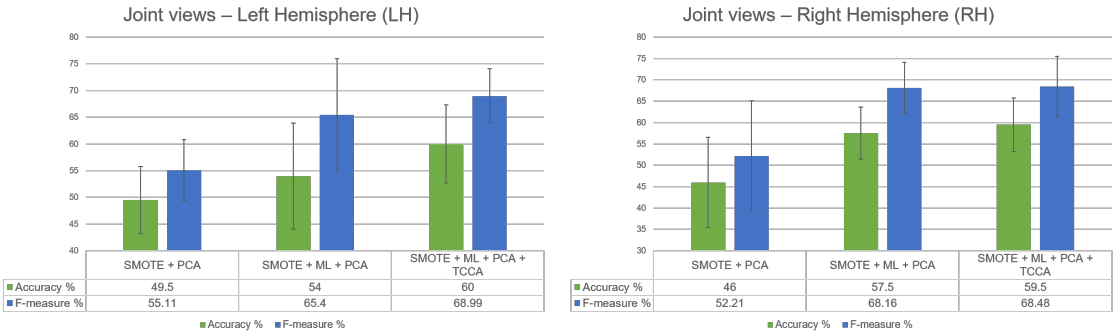


Figure 9: *Evaluation of the ASD/NC classification using the accuracy and the F-measure metrics for the left and the right hemispheres for four views.*

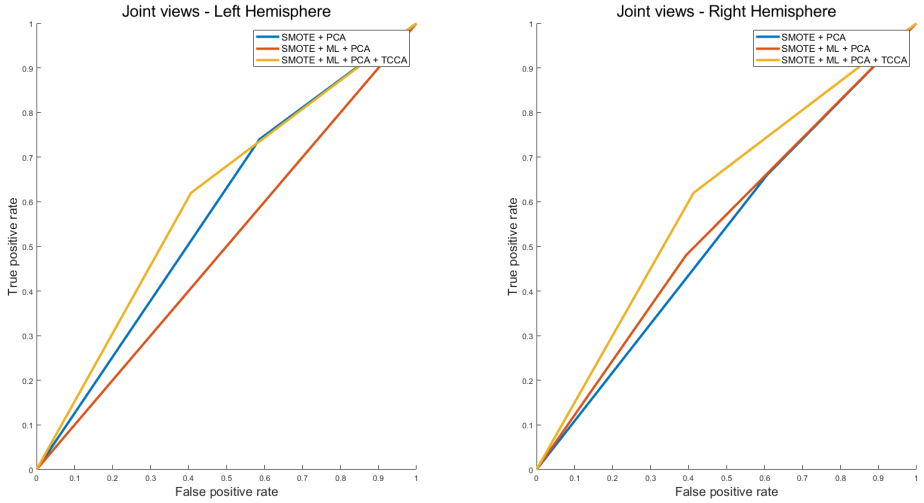


Figure 10: *ROC curves for the evaluation of the performance of the ASD/NC classification for the left and the right hemispheres using multi-view data representations.* Our method (SMOTE+ML+PCA+TCCA) is benchmarked against (SMOTE+PCA), (SMOTE+ML+PCA).

4. Discussion

In this paper, we proposed the first work on *imbalanced multi-view* brain connectomic data classification to boost the classification accuracy and avoid overfitting. The proposed MV-LEAP pipeline comprises a view-specific proliferation step and a joint view alignment step. Specifically, it first (i) *proliferates* real samples of the minority class nested in a *learned* manifold to handle imbalanced classes, then (ii) *jointly projects* both real and proliferated data views to a shared low-dimensional subspace where their distributions are aligned.

Minority sample proliferator. To evaluate the performance of our proliferator based on manifold learning (SMOTE+ML+SVM) classification framework, we evaluated it on individual brain views for both left and right hemispheres and benchmarked it against different baseline methods including: SVM (trained using original imbalanced dataset without proliferation), SMOTE+SVM, ADASYN+SVM, and ADASYN+ML+SVM. We note that SVM suffered from data overfitting as it classified well samples drawn from the majority class and failed to classify minority class samples (**Table** 3). In other words, the ASD/NC classification by SVM is biased. Hence, our proposed method is compared against remaining methods which proliferated the minority class.

Table 3: *Classification results on single views using SVM.* The sensitivity and the specificity (recall) results demonstrate that SVM predicts well the majority class, but it is very weak for predicting the minority class. Hence, the SVM classification of imbalanced data is biased. RH: right hemisphere. LH: left hemisphere. v: data view.

	LH v1	LH v2	LH v3	LH v4	RH v1	RH v2	RH v3	RH v4
Balanced accuracy %	50.34	47	52	49.34	49	51.67	51	51.34
Overall accuracy %	74.5	71.5	76	74	73.5	73.5	74.5	75
Sensitivity %	98.67	94	100	98.67	98	95.33	98	98.67
Specificity %	2	4	4	0	0	8	4	4

Considering the results of all the evaluation metrics, we notice that the two proliferators (SMOTE and ADASYN) that leveraged manifold learning to identify the k nearest neighbors in the sample synthesis step, achieved the highest classification results. This clearly demonstrated that manifold learning improved the representativeness of the synthesized samples and captured well their high-dimensional relationships.

Multi-view representations alignment. In order to evaluate the performance of our pipeline MV-LEAP and to evaluate the benefit of using

TCCA, we compared it with SMOTE+PCA and SMOTE+ML+PCA. PCA was used in order to reduce the high dimensionality of the data and prepare it for TCCA. Indeed, PCA was used for both training and testing data to enable a reliable prediction. Therefore, the dimension of the training data has decreased a lot, from 595 features to 39 features. This may lead to a decrease in the performance of the pipeline. For this reason, we compared MV-LEAP with methods that use the dimension reduction of PCA to have a fair comparison. For datasets composed of three different views, our method achieved the highest classification results using both accuracy and F-measure. SMOTE+PCA achieved the lowest overall results. As for the dataset composed of all four views, our method achieved the highest results for both the left and the right hemispheres. In summary, our proposed method MV-LEAP boosted ASD/NC classification results when using multi-view connectomic data.

Limitations and future works. Our method has a few limitations. *First*, TCCA cannot handle high-dimensional data. In fact, one needs to first reduce the dimensionality of the input data, hence the use of PCA as a preliminary step. Besides, TCCA maps and aligns all single views in one common subspace but it also reduces the dimensionality of the data. In our future work, we will seek to develop a joint multi-view data mapping and alignment method that can efficiently handle high-dimensional data. *Second*, we note that the proliferation and view alignment steps are performed in a sequential manner. In the spirit of bi-directional learning introduced in recent medical image segmentation works (Amiri et al., 2018; Bnouni et al., 2018) and which has outperformed sequential learning, we can design a proliferation-alignment bidirectional learning model, where the proliferated data improves the learning of the joint mapping transformation, and in turn the learned multi-view data mapping improves the quality of the proliferated data. This is a novel research direction that we intend to investigate in our future work building on this seminal model. *Third*, the proposed framework proliferates each data view *independently*. Ideally, one would learn how to simultaneously proliferate data views while enforcing shared traits across data domains. *Last*, our method uses a supervised classifier, which limits the scalability of the proposed classification framework. We will investigate the potential of unsupervised techniques (Yin et al., 2017) to proliferate, align, and classify unlabeled multi-view data.

5. Conclusion

In this paper, we proposed a novel framework for boosting classification performance of imbalanced multi-view connetomic data. The proposed MV-LEAP comprises two steps. The first step aims to proliferate synthetic data based on view-specific manifold learning to generate more true synthetic samples to the original data. The second step aligns both original and proliferated multi-view data into a shared subspace where low-dimensional embeddings are jointly learned. Overall, our method boosted the classification results of autistic and typical brains compared with different proliferation techniques with and without multi-view data alignment. Although the training data is *imbalanced*, *high-dimensional*, and with *overlapping classes*, MV-LEAP achieved the best ASD/NC classification accuracy using four data views and it consistently outperformed baseline methods using different evaluation metrics. In our future work, we will extend the proposed method by jointly proliferating and aligning multi-view connectomic data within a unified optimization-based framework.

Abdi, H., Williams, L.J., 2010. Principal component analysis. Wiley interdisciplinary reviews: computational statistics 2, 433–459.

Ahmad, M., Lee, S.W., 2006. Human action recognition using multi-view image sequences.

Amini, M., Usunier, N., Goutte, C., 2009. Learning from multiple partially observed views-an application to multilingual text categorization.

Amiri, S., Mahjoub, M.A., Rekik, I., 2018. Tree-based ensemble classifier learning for automatic brain glioma segmentation. Neurocomputing 313, 135–142.

Baron-Cohen, S., Ring, H.A., Wheelwright, S., Bullmore, E.T., Brammer, M.J., Simmons, A., Williams, S.C., 1999. Social intelligence in the normal and autistic brain: an fmri study. European journal of neuroscience 11, 1891–1898.

Bnouni, N., Rekik, I., Rhim, M.S., Amara, N.E.B., 2018. Dynamic multi-scale cnn forest learning for automatic cervical cancer segmentation. International Workshop on Machine Learning in Medical Imaging , 19–27.

Cao, H., Bernard, S., Sabourin, R., Heutte, L., 2019. Random forest dissimilarity based multi-view learning for radiomics application. Pattern Recognition 88, 185–197.

Capponi, C., Koço, S., 2019. Learning from imbalanced datasets with cross-view cooperation-based ensemble methods, in: Linking and Mining Heterogeneous and Multi-view Data. Springer, pp. 161–182.

Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., 2002. Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research 16, 321–357.

Chawla, N.V., Japkowicz, N., Kotcz, A., 2004. Special issue on learning from imbalanced data sets. ACM Sigkdd Explorations Newsletter 6, 1–6.

Chawla, N.V., Lazarevic, A., Hall, L.O., Bowyer, K.W., 2003. Smoteboost: Improving prediction of the minority class in boosting.

Christensen, D.L., Braun, K.V.N., Baio, J., Bilder, D., Charles, J., Constantino, J.N., Daniels, J., Durkin, M.S., Fitzgerald, R.T., Kurzius-Spencer, M., et al., 2018. Prevalence and characteristics of autism spectrum disorder among children aged 8 yearsautism and developmental disabilities monitoring network, 11 sites, united states, 2012. MMWR Surveillance Summaries 65, 1.

Comon, P., Luciani, X., De Almeida, A.L., 2009. Tensor decompositions, alternating least squares and other tales. Journal of Chemometrics: A Journal of the Chemometrics Society 23, 393–405.

Desikan, R.S., Ségonne, F., Fischl, B., Quinn, B.T., Dickerson, B.C., Blacker, D., Buckner, R.L., Dale, A.M., Maguire, R.P., Hyman, B.T., et al., 2006. An automated labeling system for subdividing the human cerebral cortex on mri scans into gyral based regions of interest. Neuroimage 31, 968–980.

Dhifallah, S., Rekik, I., Initiative, A.D.N., et al., 2019. Clustering-based multi-view network fusion for estimating brain network atlases of healthy and disordered populations. Journal of neuroscience methods 311, 426–435.

Fernández, A., Garcia, S., Herrera, F., Chawla, N.V., 2018a. Smote for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary. Journal of Artificial Intelligence Research 61, 863–905.

Fernández, A., Garcia, S., Herrera, F., Chawla, N.V., 2018b. Smote for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary. Journal of Artificial Intelligence Research 61, 863–905.

Fischl, B., 2012. Freesurfer. Neuroimage 62, 774–781.

Guo, Y., Xiao, M., 2012. Cross language text classification via subspace co-regularized multi-view learning. arXiv preprint arXiv:1206.6481 .

Han, H., Wang, W.Y., Mao, B.H., 2005. Borderline-smote: a new over-sampling method in imbalanced data sets learning.

He, H., Bai, Y., Garcia, E.A., Li, S., 2008. Adasyn: Adaptive synthetic sampling approach for imbalanced learning.

Kroonenberg, P.M., De Leeuw, J., 1980. Principal component analysis of three-mode data by means of alternating least squares algorithms. Psychometrika 45, 69–97.

Li, D.C., Hu, S.C., Lin, L.S., Yeh, C.W., 2017. Detecting representative data and generating synthetic samples to improve learning accuracy with imbalanced data sets. PloS one 12, e0181853.

Li, X., Liu, Q., He, Z., Wang, H., Zhang, C., Chen, W.S., 2016. A multi-view model for visual tracking via correlation filters. Knowledge-Based Systems 113, 88–99.

Li, Y., Yang, M., Zhang, Z.M., 2018. A survey of multi-view representation learning. IEEE Transactions on Knowledge and Data Engineering .

Lisowska, A., Rekik, I., disease neuroimaging initiative (ADNI), T.A., 2018. Joint pairing and structured mapping of convolutional brain morphological multiplexes for early dementia diagnosis. Brain connectivity 9, 22–36.

Lisowska, A., Rekik, I., Initiative, A.D.N., et al., 2017. Pairing-based ensemble classifier learning using convolutional brain multiplexes and multi-view brain networks for early dementia diagnosis.

Liu, M., Zhang, J., Yap, P.T., Shen, D., 2017. View-aligned hypergraph learning for alzheimers disease diagnosis with incomplete multi-modality data. Medical image analysis 36, 123–134.

Liu, X.Y., Wu, J., Zhou, Z.H., 2009. Exploratory undersampling for class-imbalance learning. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 39, 539–550.

Luo, S., Zhang, C., Zhang, W., Cao, X., 2018. Consistent and specific multi-view subspace clustering.

Luo, Y., Liu, T., Tao, D., Xu, C., 2015a. Multiview matrix completion for multilabel image classification. IEEE Transactions on Image Processing 24, 2355–2368.

Luo, Y., Tao, D., Ramamohanarao, K., Xu, C., Wen, Y., 2015b. Tensor canonical correlation analysis for multi-view dimension reduction. IEEE transactions on Knowledge and Data Engineering 27, 3111–3124.

Ma, L., Fan, S., 2017. Cure-smote algorithm and hybrid algorithm for feature selection and parameter optimization based on random forests. BMC bioinformatics 18, 169.

Mahjoub, I., Mahjoub, M.A., Rekik, I., 2018. Brain multiplexes reveal morphological connectional biomarkers fingerprinting late brain dementia states. Scientific reports 8, 4103.

Maloof, M.A., 2003. Learning when data sets are imbalanced and when costs are unequal and unknown.

Nebli, A., Rekik, I., 2019. Gender differences in cortical morphological networks. Brain Imaging and Behavior .

Provost, F., 2000. Machine learning from imbalanced data sets 101.

Raeper, R., Lisowska, A., Rekik, I., 2018. Cooperative correlational and discriminative ensemble classifier learning for early dementia diagnosis using morphological brain multiplexes. IEEE Access 6, 43830–43839.

Ramentol, E., Verbiest, N., Bello, R., Caballero, Y., Cornelis, C., Herrera, F., 2012. Smote-frst: a new resampling method using fuzzy rough set theory, World Scientific, pp. 800–805.

Roffo, G., Melzi, S., Cristani, M., 2015. Infinite feature selection.

Serra, A., Galdi, P., Tagliaferri, R., 2019. Multiview learning in biomedical applications.

Shao, K., Zhai, Y., Sui, H., Zhang, C., Ma, N., 2014. A new over-sample method based on distribution density. JCP 9, 483–490.

Soussia, M., Rekik, I., 2017. High-order connectomic manifold learning for autistic brain state identification. International Workshop on Connectomics in Neuroimaging , 51–59.

Soussia, M., Rekik, I., 2018. Unsupervised manifold learning using high-order morphological brain networks derived from t1-w mri for autism diagnosis. Frontiers in Neuroinformatics 12.

Sporns, O., 2013. Structure and function of complex brain networks. Dialogues in clinical neuroscience 15, 247.

Sun, S., 2013. A survey of multi-view machine learning. Neural Computing and Applications 23, 2031–2038.

Sun, Y., Wong, A.K., Kamel, M.S., 2009. Classification of imbalanced data: A review. International Journal of Pattern Recognition and Artificial Intelligence 23, 687–719.

Wang, B., Ramazzotti, D., De Sano, L., Zhu, J., Pierson, E., Batzoglou, S., 2017. Simlr: a tool for large-scale single-cell analysis by multi-kernel learning. bioRxiv , 118901.

Wang, H., Feng, J., Zhang, Z., Su, H., Cui, L., He, H., Liu, L., 2018. Breast mass classification via deeply integrating the contextual information from multi-view data. Pattern Recognition 80, 42–52.

Wang, Z., Zhu, Y., Chen, Z., Zhang, J., Du, W., 2019. Multi-view learning with fisher kernel and bi-bagging for imbalanced problem. Applied Intelligence , 1–14.

Wu, F., Jing, X.Y., You, X., Yue, D., Hu, R., Yang, J.Y., 2016. Multi-view low-rank dictionary learning for image classification. Pattern Recognition 50, 143–154.

Yang, M., Deng, C., Nie, F., 2019. Adaptive-weighting discriminative regression for multi-view classification. Pattern Recognition 88, 236–245.

Yin, Q., Wu, S., Wang, L., 2017. Unified subspace learning for incomplete and unlabeled multi-view data. Pattern Recognition 67, 313–327.

Zhang, J., Liu, M., Lu, K., Gao, Y., 2019. Group-wise learning for aurora image classification with multiple representations. IEEE transactions on cybernetics .

Zhao, J., Xie, X., Xu, X., Sun, S., 2017. Multi-view learning overview: Recent progress and new challenges. Information Fusion 38, 43–54.

Zhao, Y., You, X., Yu, S., Xu, C., Yuan, W., Jing, X.Y., Zhang, T., Tao, D., 2018. Multi-view manifold learning with locality alignment. Pattern Recognition 78, 154–166.